

**In the Specification:**

A substitute Specification accompanies this Response.

On page 1, immediately following the CROSS REFERENCE TO RELATED APPLICATIONS heading, please replace Paragraph **[0001]** with the text as shown:

**[0001]** The present application is related to commonly assigned, co-pending U.S. patent application No. 09/794,302, filed on February 27, 2001, entitled METHOD AND APPARATUS FOR VIEWING ELECTRONIC COMMERCE-RELATED DOCUMENTS, by inventors Andrew Everett Davidson, Kelly Lane Schwarzhoff, Gunawan Herri, Changyi Zhu, Ari Krish, Muljadi Sulistio, and Sun Keun Lee, which is hereby incorporated by reference as if fully set forth.

On page 1, please replace Paragraph **[0002]** with the text as shown:

**[0002]** This application is further related to the following commonly assigned, co-pending applications filed on December 18, 2001, the same day as the present application: U.S. patent application **10/026,663**, xCBL MAILBOX METHODS AND DEVICES, by inventors Muljadi Sulistio, Yang Wei, Kelly Lane Schwarzhoff, Yuan Din, Sun Lee and Andy Yang; U.S. patent application No. **10/026,364**, METHOD AND APPARATUS FOR DECLARATIVE UPDATING OF SELF-DESCRIBING, STRUCTURED DOCUMENTS, by inventors Muljadi Sulistio, Yang Wei, Kelly Lane Schwarzhoff and Yuan Din; and U.S. patent application No. **10/026,366**, METHOD AND APPARATUS FOR DECLARATIVE ERROR HANDLING AND PRESENTATION, by inventors Muljadi Sulistio, Yang Wei, Kelly Lane Schwarzhoff and Yuan Din. These applications filed on the same day are hereby incorporated by reference as if fully set forth.

On page 2, immediately following the COPYRIGHT NOTICE heading and Paragraph **[0003]**, please insert new Paragraph **[0004]** with the text as shown:

**REFERENCE TO COMPUTER PROGRAM LISTING APPENDIX**

**[0004]** A computer program listing appendix comprising duplicate copies of a compact disk, named "JGR 1008-1 CPLA" accompanies this application and is incorporated by reference. The computer program listing appendix includes the following files:

Default_xsl1.txt	9 Kbytes on disk	created 21 March 2005
Default_xsl2.txt	7 Kbytes on disk	created 21 March 2005
JSP_CODE.txt	22 Kbytes on disk	created 21 March 2005

On page 7, please replace Paragraph **[0040]** with the text as shown:

**[0040]** The document type selection screen of figure 4 can lead to either a search or a reply action, depending on the context. In figure 4, a trading partner name and document type may be described in the banner 441, if they have been selected. Selection of a document type provides a context for either searching or replying.

On page 7, immediately following Paragraph **[0040]**, please add new Paragraph **[0040A]** as shown:

**[0040A]** Figure 5 depicts a sample search screen that includes many of the same features as other sample screens, e.g., 211, 212, 221, and 331. One or more field selection filters 512, 513 are supplied. The values accessible by the pull down menus of these filters are context sensitive to a selected document type, if the document type is selected by a filter 518 or has been selected by context in a prior screen. Only the field types that are valid for a particular document type appear in the pull down menu, once a document type has been selected. One or more value specification fields 514 also are provided. These value specification fields are context sensitive to the document fields. The user's entry can be checked for format and data type as the user enters the data. The combination of field selection filters and value specification fields, context sensitive to the document type selected, enables a generic interface for

searching across different document types. In addition to generic field handling, additional filters can be provided to search by date 515, document status 516, trading partner 517 or other field. A variety of interfaces can be used to implement these filters, such as pull down menus, radio buttons, checkboxes, or blanks to fill in. A list of folders to search in 519 can be dynamically generated, to match the list of folders in the navigation box 221, 331.

On page 9, please replace Paragraph **[0045]** with the following text:

**[0045]** On server 708, one or more services 711-713 may be available. One service may receive the document 711. The same or another service 712 may persistently store the incoming document, for instance in a database 707. One or more databases may be used to store data useful for electronic commerce or other document exchange. A database may include a repository of schemas 741 for standard and entity-defined business documents, a repository of JavaBeans 742, C++ structures, Pascal records or scripts useful to electronic commerce, a document map repository 743 for translation of documents from one format to another (e.g., the xCBL format to an export format) and for transformation of documents from one type to another (e.g., from a request for quotation to a quotation.) The database may further include a report layout repository 744 and a presentation layout repository 745. The presentation layout repository may include declarative transformations for changing documents from a Sox document format to an HTML format, and back again. The transmission properties data describes the transport information needed to sent a business document to a recipient. For example, SSL security credentials can be stored as transmission properties. A trading partner directory is also useful. The trading partner directory may, as described above, identify URLs to which the sending entity 701 transmits documents. The persistent storage for data 707, 741-47 may be on a single data storage unit or multiple units. Persistent document storage need not be part of a database. An indexed flat file would suffice to store XML-compliant documents. The services host 708 may also host an indexing service to index one or more of the tagged fields. Alternatively, a database system managing the persistent storage or other

subsystem may index documents for retrieval. An incoming document may or may not be validated against schema from a schema repository 741, before it is persistently stored. A schema may be used to interpret the document. One or more JavaBeans or scripts may be used to act on the document before it is stored.

On page 14, please replace Paragraph **[0052]** with the following text:

**[0052]** Sample excerpts of a document to display transformation 1212 for a main processing routine and display of header information with embedded path specifications are reflected in the file "Default\_xsl1.txt" on the accompanying CD-ROM of source code.

On page 21, please replace Paragraph **[0054]** with the following text:

**[0054]** Sample excerpts of a document to document transformation 1211 from a request for quotation to quotation follows, including portions of a main processing routine and a party copy routine, are reflected in the file "Default\_xsl2.txt" on the accompanying CD-ROM of source code.

On page 25, please replace Paragraph **[0056]** with the following text:

**[0056]** Excerpts from a sample of a user form 141 that produced the sample screen in Figure 13 are reflected in the file "JSP\_CODE.txt" on the accompanying CD-ROM of source code.

On page 51, please amend the **Abstract** with the text as shown:

The present invention includes a method and device for generic searching across document types. A further aspect of the present invention is indexing and searching XML documents using an HTML interface generated by a declarative transformation.